Environment Canada
Conservation and Protection

Environnement Canada
Conservation et protection

National Hydrology Research Institute
Institut national de recherche en hydrologie

Inland Waters and Lands Directorate
Saskatoon, Saskatchewan, Canada

Direction générale des eaux intérieures et les terres
Saskatoon, Saskatchewan, Canada

Canada

ISSN 0838-1992

A. Vandenberg

Ground Water Division
National Hydrology Research Institute
Environment Canada
Saskatoon, Saskatchewan
S7N 3H5

A DIGITAL SIMULATION OF MOISTURE MOVEMENT THROUGH A LAYERED COLUMN OF SAND AND AFBC SOLID WASTE

PRELIMINARY REPORT, FEBRUARY 1987

NHRI Contribution No.88015

April, 1988

MANAGEMENT PERSPECTIVE

A large Facility for Indoor Aquifer Testing (FIAT) is being constructed at the National Hydrology Research Institute (NHRI) to permit the study of moisture and contaminant movement in a ground-water system under controlled conditions. The model documented in this paper simulates the flow in the soil/ground-water system in the tank. The model can be used to assess the feasibility of using such a tank to simulate flow through a disposal site and to predict the course of the experiments to be undertaken in the tank. The ability to predict contaminant migration and then to verify these predictions will increase the understand of the contaminant migration process and aid in pollution control.

PERSPECTIVE DE GESTION

On procède en ce moment à l'Institut national de recherche en hydrologie (INRH) à la construction d'une grande cuve permettant de faire des essais sur nappe aquifère à l'intérieur; cette installation permettra d'étudier la migration de l'humidité et des contaminants dans un système d'eau souterraine dans des conditions contrôlées. Le modèle visé dans ce rapport simule l'écoulement dans le système sol/eau souterraine reproduit dans la cuve; il peut servir à déterminer s'il est possible d'utiliser une cuve pour simuler l'écoulement dans un site d'enfouissement et à prédire le déroulement des expériences qui vont y être menées. Si l'on avait la capacité de déterminer la migration des contaminants et, par la suite, de vérifier si les prédictions étaient exactes, on comprendrait alors mieux le processus de migration des contaminants et on pourrait plus facilement combattre la pollution.

ABSTRACT

A digital model was constructed to simulate the vertical movement of soil moisture through an unsaturated bed of Atmospheric Fluidised Bed Combustion (AFBC) solid waste, underlain and overlain by sandy soil. The model is based on a finite difference solution to Richard's equation and features:

1) an iterative routine for the calculation of flov between adjacent elements, when the elements are soil-physically dissimilar, and

2) a self-adjusting time step.

The model and two examples of a simulation are discussed. Details on the use of the program, a FORTRAN source code listing, and an example of a data set are given in the appendices. A diskette with the necessary files for running the program on an IBM PC-XT computer is available on request.

RÉSUMÉ

Un modèle digital a été reproduit pour simuler le mouvement vertical de l'humidité du sol dans un lit non saturé de déchets solides provenant de la combustion atmosphérique sur le lit fluidisé (AFBC) et se trouvant entre deux couches de sol sablonneux. Le modèle est basé sur une solution en différences finies à l'équation de Richard et comprend

1) une méthode itérative pour le calcul de l'écoulement entre des éléments adjacents, lorsque les propriétés physiques du sol diffèrent, et

2) un pas de temps auto-réglable.

Le modèle et deux exemples de simulation sont traités. Des détails sur l'utilisation du programme, une liste en langage source FORTRAN et un exemple d'ensemble de données figurent aux appendices. On peut obtenir sur demande une disquette renfermant les dossiers nécessaires pour exécuter le programme sur un ordinateur IBM PC-XT.

TABLE OF CONTENTS

Page

Introduction 1

Brief description of the numerical model 3

Preliminary test runs 6

Conclusion 8

References 11

Appendices . . . . .

A. TANKHELP, a file with helpful hints for the user of program TANKFLO 15

B. FORTRAN source code 31

C. The example data set: file TANKTEST.DAT 41

Tables

1. Estimated physical properties of the soil 12

2. Estimated physical properties of the waste 13

3. Results of the model application: Equilibrium moisture distribution at three rates of throughflow 14

INTRODUCTION

A large facility for indoor aquifer testing (FIAT) is being built at the National Hydrology Research Institute (NHRI) for the simulation of moisture movement and leaching of toxic components in a disposal site for Atmospheric Fluidized Bed Combustion (AFBC) solid waste. The testing facility is a cylindrical, fiberglass tank, 4 m high and 3 m in diameter (Carter, 1987). The bottom 2 m of the tank will be filled with the sandy soil used at the actual disposal site. On this sand layer a 1 m thick layer of the waste will be deposited and in turn overlain by 1 m of sand. It is expected that the bottom of the sand/waste column can be maintained at atmospheric pressure, but may at times be allowed to build up pressure by restricting the outflow to a constant rate.

Moisture input to the model will be from an overhead sprinkler, distributing the moisture evenly over the surface of the column. Plans call for a maximum application rate of 20 cm/day.

In order to test the feasibility of using such a tank to simulate flow through a disposal site, a numerical model has been constructed based on an existing numerical model for the simulation of moisture movement in drained fields (Vandenberg, 1985; Vandenberg, 1987).

Both the original and the derived model use the same method for the simulation of flow in the interior of the soil column, but the boundary conditions imposed by the tank application are different, and extensive modifications were required.

The purpose of this report is to document the model, to describe the limited testing of the model carried out at this time, and to give instructions for its use. In the following sections the model is first discussed very briefly; this is followed by a description of two simulation runs, based on assumed soil-physical data for the sand and the waste. Finally a few problems that were found during the testing of the model are reviewed. The appendices give necessary information for the operation of the model.

BRIEF DESCRIPTION OF THE NUMERICAL MODEL

In the numerical model the soil column is subdivided into a number of thin soil elements. At set times, separated in time by a small time period or timestep, Δt, the flow rate from one element to the next physically higher element is calculated for all the elements, using Richard's equation. For each element, the total increase in moisture content during a timestep is calculated as the difference between the inflow rate to that element from the element below and the outflow rate to the element above, multiplied by the length of the timestep. Both the inflow rate and the outflow rate may be negative indicating downward moisture flow. The uppermost element receives inflow from the element below, and infiltration from above from water pooled on the surface. The pooled water results from precipitation. The lowermost element loses water to, or gains water from, the element above it; it may also lose or gain water through its lower surface, either collected by, or supplied from a storage element called "ditch". This flow rate into or out of the column through its lower boundary is technically known as a boundary condition. It may be specified as a rate which remains constant over a period of time, or it may be calculated using Richard's equation and assuming that the moisture content of the lowermost element is kept constant. In the first instance the boundary condition is a so called constant flux boundary condition; in the latter case it is a constant potential boundary condition.

For the evaluation of flow between elements using Richard's equation, an extensive set of soil physical data is needed for both the sand and the waste. These must be given as input to the program, summarized from the available data, in two tables: one table giving the relation between soil moisture, θ, and pressure, ϕ, and the other table giving the relation between θ and the conductivity, K. At the time of writing of this report these data were not yet available, and a set of data typical of an average sand (Beuving, 1984) has been used temporarily. For the waste the only available datum was the permeability at saturation, which was reported to range from 6.8×10^{-4} to 3.7×10^{-2} cm/s (60 to 3000 cm/day); a value of 80 cm/day has been used temporarily to test the model. This value was extended into the unsaturated range by an exponential equation of the form:

$$K = K_0 \exp(\alpha\phi)$$

since the $K-\phi$ relation has often been observed to be of this form over part of the unsaturated range. The α was chosen as 0.02. The $\theta-\phi$ relation was constructed on the assumption that the waste is highly hygroscopic and reluctant to release water unless near saturation.

The soil-physical characteristics of the soil and the waste, estimated in this manner, are given in Table 1 and Table 2. They form the bulk of the input data. Other data, such as the length of the timestep and the number and thickness of the elements, will be discussed in detail in Appendix A.

This data set was used in debugging the program, and, with debugging completed, provided the basis for some test-runs. These tests tend to show what might happen when the tank is put into operation. Some of the details of these tests are provided in the next section.

PRELIMINARY TEST RUNS

In the first test 20 cm/day of precipitation was applied to a soil and waste profile originally at equilibrium with no internal water flow and with the bottom element at atmospheric pressure and saturated. This means that the elements near the top of the tank are very dry (table 3, column 2). Since the pressure must decrease from top to bottom by an amount equal to the height of the soil column, this condition gives a pressure of -380 cm for the top element, which implies that the conductivity there is only 0.0002 cm/day. In the uppermost element of the waste the pressure is -280 cm and the conductivity is 0.7 cm/day. As a consequence, the applied water is mainly absorbed in the upper soil elements at first. Only after 0.8 days is there a noticeable increase in the moisture content of the sand element just above the waste. The upper waste element starts to moisten at the end of the first day, and at the end of the second day it has reached its equilibrium moisture content for that precipitation rate. The outflow from the bottom remains negligible for 2.6 days, at which time the outflow increases very rapidly to attain equilibrium flow (20 cm/day) within the hour. The equilibrium moisture distribution at this rate is shown in table 3, column 4.

After equilibrium was established throughout the column, the precipitation was discontinued. With the bottom of the column still at constant atmospheric pressure and saturated, the outflow remained high for about one hour, then dropped sharply to approximately 8 cm/day over the next 2 hours. The outflow continued to drop slowly to 3.5 cm/day one full day after precipitation ended; and to 2.3 cm/day after two full days.

In the next test the same precipitation rate was applied. This time the column was not at equilibrium with zero throughflow at the start of the simulation, but at the near equilibrium flow which had established itself at the end of the first test. With the moisture moving downward at the rate of 2.3 cm/day at the bottom of the tank, the soil elements are much wetter than at the start of the previous test (table 3, column 3). As a result the outflow from the bottom started to increase after 0.564 days, and equilibrium flow was established after 0.865 days.

CONCLUSION

The mechanics of the model have been tested under a number of alternative situations, and do not present any major problems. These tests are not reported here, but included the use of a less permeable soil, and varying the timestep, the number and thickness of the elements, and the precipitation rate. Some minor problems, which are inherent in the application of the model, cannot be eliminated without major technical and mathematical improvement, if at all.

One such problem is the time it takes to simulate a few days of tank-operation. Under saturated or near saturated conditions the maximum timestep which can be used becomes very small due to the higher conductivity and especially due to the high rate of change, $d\phi/d\theta$. A possible remedy here is to provide more detailed data on the ϕ-θ curve in the near saturated range and to ensure that the increase in $d\phi/d\theta$ occurs gradually. This does not necessarily mean that more detailed laboratory data must be provided in the near saturated range. But the available data must be interpolated at smaller intervals and using a smooth interpolation curve. Some experimentation will be needed for each simulation to determine the number of interpolated data points required for optimum results.

Another area which gives problems is the establishment of a proper initial condition. Indications are that only steady state with zero throughflow is an error free initial condition. If one tries to prescribe an arbitrary initial moisture profile, which is one of the options, the program may have difficulty establishing inter-elementary flow rates at boundaries between elements with dissimilar soil-physical properties. This difficulty is caused both by the finite approximation in the model to an essentially infinitesimal process, and to the incompleteness of and the uncertainties inherent to the moisture data which are used as input.

On the other hand, if one tries to let the program calculate initial conditions from a prescribed steady state throughflow, which is another option of the program, difficulties may be encountered in establishing the pressure changes between neighbouring elements of dissimilar properties. However, in both cases the time-dependent part of the program seems to recuperate sufficiently from these "unrealistic" initial conditions and to give "realistic" values after an initial period of "breaking in". In this case, the output from the first run can be used as the "realistic" initial condition for a subsequent "real problem" run. The results of such a run are given in the second example of the previous section.

The numerical results obtained in the two examples are of course entirely dependant on the actual soil-physical properties of the soil and waste. Since both the sand and the waste may have conductivities over a range of several orders of magnitude, actual results may differ from those presented by the same amount. Therefore, the examples should only be taken as an indication of the kind of predictions the model can produce. It should also be understood that very coarse estimates of, for example, the rate of throughflow under equilibrium conditions at a given precipitation rate, can be obtained from the conductivity of the materials alone, by a quick calculation "on the back of an envelope" and no elaborate model is needed.

The program is useful for the calculation of the time it takes for a wetting front to reach the waste, or the time it takes for equilibrium flow to be established. In other words, the model is useful whenever the unsteady nature of the moisture movement is important, as in determining flow through the soil column contained in the large scale tank.

REFERENCES

BEUVING, J., 1984. Vocht- en doorlatendheids karakteristieken, dichtheid en samenstelling van bodemprofielen in zand-, zavel-, klei- en veengronden. (Moisture and conductivity characteristics, density and composition of soil profiles in sandy, loamy, clayey and peaty soils) .26 p. Report #10, ICW, P.O Box 35, 6700 AA, WAGENINGEN, The Netherlands.

CARTER, T., 1987. Facility for Indoor Aquifer Testing: Design Report, Progress to March 31, 1987. NHRI Unpublished Report, Contribution No. 87006, 34 pp.

VANDENBERG, A., 1985. A physical model of vertical infiltration, rain discharge and surface runoff. NHRI Paper No. 25, National Hydrology Research Institute, Saskatoon, (Saskatchewan), 57 pp.

VANDENBERG, A., 1987. A physical model of infiltration, drain discharge and surface runoff for layered soils. NHRI Paper No. 42, National Hydrology Research Institute, Saskatoon, (Saskatchewan), in publication.

TABLE 1

ESTIMATED PHYSICAL PROPERTIES OF THE SOIL

θ Soil Moisture (% by volume)	K Conductivity (cm/day)	ϕ Pressure (cm)
.50	.0000001	-15850
.52	.0000010	-10000
.54	.0000012	-7943
.57	.0000015	-5623
.60	.000004	-3162
.65	.000011	-1778
.70	.000031	-1000
.75	.000046	-794
.90	.000085	-562
1.7	.000234	-316
3.7	.00105	-177
9.2	.1905	-100
22.6	1.449	-56.2
24.2	2.366	-31.6
24.8	8.00	-10.0
25.0	27.54	-3.16
25.1	39.88	0.0
25.11	39.88	20.0

TABLE 2

ESTIMATED PHYSICAL PROPERTIES OF THE WASTE

θ Soil Moisture (% by volume)	K Conductivity (cm/day)	ϕ Pressure (cm)
5.0	0.000001	-15850
9.0	0.00001	-10000
9.5	0.00004	-7943
10.8	0.00007	-5623
13.9	0.0001	-3162
17.0	0.0002	-1778
20.7	0.0005	-1000
22.3	0.0008	-794
24.5	0.0011	-562
27.4	0.144	-316
30.0	2.32	-177
32.5	10.83	-100
34.0	26.00	-56.2
35.5	42.5	-31.6
37.6	65.5	-10.0
37.8	75.1	-3.16
37.9	80.0	0.0
37.91	80.0	20.0

TABLE 3

RESULTS OF MODEL APPLICATION:

EQUILIBRIUM MOISTURE DISTRIBUTION, AS VOLUME RATIO'S,

AT THREE RATES OF THROUGHFLOW

Elem.#	Flowrate = 0 (cm/day)	Flowrate =-2.3 (cm/day)	Flowrate =-20 (cm/day)
20(top)	.0149	.1609	.2492
19	.0156	.1913	.2493
18	.0162	.2035	.2492
17	.0169	.1965	.2493
16	.0193	.1476	.2465
	-------- top of waste --------		
15	.2807	.3211	.3423
14	.2845	.3264	.3466
13	.2882	.3323	.3520
12	.2920	.3384	.3609
11	.2957	.3485	.3733
	------- bottom of waste -------		
10	.0366	.2414	.2492
9	.0491	.2417	.2492
8	.0634	.2419	.2492
7	.0777	.2419	.2492
6	.0920	.2420	.2493
5	.1532	.2420	.2489
4	.2144	.2421	.2442
3	.2365	.2431	.2383
2	.2452	.2458	.2453
1	.2510	.2510	.2510

APPENDIX A.

TANKHELP, A FILE WITH HELPFUL HINTS FOR THE USER OF PROGRAM TANKFLO.

This appendix is a copy of file TANKHELP, which has been placed on diskette together with the files needed to run program TANKFLO. TANKFLO is the name of the digital model described in this report.

INTRODUCTION

TANKHELP describes the latest version of program TANKFLO, dated Feb. 6, 1987. TANKFLO is an adaptation of program DRAIN (Vandenberg,1987), specially conceived to simulate moisture movement in the Facility for Indoor Aquifer Testing (FIAT), a large soil-test tank being built at NHRI, Saskatoon. The program will therefore omit the calculation of drain-discharge using Hooghoudt's relation, as used in program DRAIN. Instead it calculates drain outflow from the bottom element by assuming that the bottom element is at atmospheric pressure and saturated over a period of time. Alternatively, the outflow from or inflow to the bottom element may be specified as being constant over a period of time. During execution the user is prompted to specify the duration of the next period, the precipitation rate over the period, and the boundary condition at the bottom of the column.

SYSTEM REQUIREMENTS

The minimum hardware required to use the files on this diskette and to execute program TANKFLO consists of an IBM PC or compatible with 64 K of RAM, and a printer. The program was developed on an IBM PC XT under the DOS operating system, using FORTRAN 2.00 by Microsoft Corp.

SOFTWARE REQUIREMENTS

Program TANKFLO and auxiliary files are available on diskette. The only file from this diskette needed to run the program is the executable file TANKFLO.EXE. For demonstration an example of an input data file, TANKFLO.DAT, has been placed on the diskette. And for the experimenter the FORTRAN source code, TANKFLO.FOR, has been included. The file TANKHELP contains documentary text giving detailed instructions on how to assemble the data file, how to run the program, and how to interpret the printed or displayed output. This file is attached to a report on the model (Vandenberg,1987) as Appendix A.

ORGANIZATION OF DATA FOR THE PROGRAM

For its input and output requirements the program uses: the standard input device (the keyboard) for interactive input of parameters; the standard output device (the screen) for output of results; and one disk file for the bulk of the input data.

However, under DOS, the standard input and output files may be redirected, e.g. output may be redirected to the printer or saved on a disk file. Or input which is supposed to be entered from the console may be redirected to come from a disk file. Details on redirection can be found in the DOS user's guide and the DOS manual.

THE COMMAND LINE: HOW TO RUN THE PROGRAM

The standard input and output devices are the program's default files, and do not have to be mentioned by the user, unless redirection is required. The name of the disk file must be passed to the program on the command line, e.g. when the system is requested to run the program by typing the file name of the executable file, e.g. the command

```
TANKFLO TEST
```

will run the program, and establishes the file TEST as the data file. Since there is no redirection, the output will be displayed on the screen, and so will the prompts for additional parameters. The responses to the prompts are to be typed on the keyboard.

If, however, the file is not mentioned on the command line the program is forgiving enough to prompt for it by displaying:

```
unit #1 ?
```

and the file name can be supplied by typing:

```
TEST
```

If redirection of the output file to the printer is required the command line would be

```
TANKFLO TEST > PRN
```

or if the output file is to be saved as a disk file named RESULT:

```
TANKFLO TEST > RESULT
```

DETAILED DESCRIPTION OF THE INPUT FILE

The input data file is in standard ASCII characters and may be prepared using the DOS editor EDLIN or an editor of your choice. For data files EDLIN is the author's choice, mostly because identical data fields on successive lines can be easily copied using the DOS editing keys and the ---> and <--- cursor keys (see the DOS user's guide).

Data are organised in the file as sequential records. A record may consist of more than one line of text, but a new record must begin on a new line. Lines may contain up to 225 characters, each line terminated by a carriage return. It is good practice to limit line length to about 70 characters, for easy printing or viewing on the screen.

The data records and individual data items on the input file are described next. For more detail on some of the parameters refer to Vandenberg (1987).

Record 1: First title line, 79 characters maximal. This one line record may contain any character information useful in identifying the application.

Record 2: :Second title line, 79 characters maximal.

Record 3: :one parameter:

NUMSIM = an integer giving the number of runs. In the conversational mode NUMSIM will usually be set to 1, only one application. The following text is based on that assumption.

Record 4: 14 parameters, separated by blanks or commas:

N = an integer, the number of nodes/elements ; up to 50 nodes may be specified.

DZ = thickness of the nodes; note that the user is responsible for ensuring that all units are consistent; there is no other constraint on the choice of systems. A suitable consistent system of units for unsaturated flow is:
Time in days
Length in cm, and therefore
Rate of flow in cm/day
Conductivity in cm/day.

NMAX = integer, maximum number of iterations allowed in the calculation of initial moisture distribution. For a detailed description, see Vandenberg (1987). NMAX = 10 will work well in most applications.

UTEST = tolerance in the iteration for initial moisture: a ratio, 0.001 is reasonable.

DPRINT = time interval at which results are written. Since the timestep may be very small, printing the results every timestep will not be needed. Set this parameter to the time interval at which you want results displayed: every day, every 1/2 day etc. Remember to use units which are consistent with the other units containing a time dimension!

DT = timestep for calculation of the moisture budget, but the supplied value will be adjusted if not in the range of suitable time steps which will be calculated by the program.

TIME = starting time of the process, usually 0.

MT = integer, number of distinct soil-types ,maximum of 5 soil types.

MTE = integer, number of table entries in the tables of θ,K, and ϕ, maximum is 50 entries.

POOL = initial height of pooled water, e.g. any water pooled on the surface of the column.

DITCH = initial height of water in ditch. DITCH does not affect the working of the program, it accumulates the outflow from the bottom element; it is usually set to 0.

IT = integer, maximum number of iterations in the calculation of flow between dissimilar elements. 10 will do nicely.

DOWNRATE= lower range of permissable timestep duration; expressed as a fraction of the maximum timestep calculated; 0.5 will do well.

UPRATE = upper bound on permissable timestep duration; expressed as fraction of the maximum timestep calculated; try 0.9 for a start. If the going timestep lies between

DOWNRATE*(max.calc.steptime) and
UPRATE*(max.calc.steptime)

the timestep will not be changed; else, it will be reset to

UPRATE*(max.calc.steptime).

Record 5: 3 integer parameters, IOPT(i),i=1 to 3;

IOPT(1) :selects the initial state of the soil-column:

= 1, then the constant outflow from the bottom element must be entered later on (L/T);

= 2, then initial pressures, ϕ, must be given for each node;
= not 1 or 2, then initial moisture content, θ, must be given for each node;

IOPT(2) :has no meaning in this version and is only kept to make the data set compatible with data sets for program DRAIN. Set to any value.

IOPT(3) = 1, then the tables representing the θ,K,and ϕ relations of the soils will be printed;
= any value but 1, then the θ-K and θ-ϕ tables will not be printed.

Record 6: N integers, IND(i),i=1 to N: for each node an integer between 1 and MT, indicating the type of soil for that element.

Record 7: IDISP and WETP(i= 1 to IDISP);
IDISP= integer, the number of nodes for which moisture content will be plotted at the time-intervals selected by DPRINT;
WETP(i): integers, the node numbers for which ϕ will be displayed.

Record 8: TABLE(i,j,k), i=1 to MTE
j=1 to 3
k=1 to MT.
This is the table of θ, K and ϕ values characterizing the MT soil types. The values must be entered in this order: MTE values of θ, then MTE values of K, then MTE values of ϕ, all for soil type #1; repeat the same format for soil type #2, etc. until all MT soil types have been entered.

Data may be continued on a new line at any point between data items, the carriage return is recognized as a separator, and the data on the new line will be recognized as belonging to the same record. A carriage return must however occur at the end of the record.

Record 9: Either QS,the initial steady state throughflow (L/T), if IOPT(1) was specified as 1;
or $\phi(i)$, i=1 to N, if IOPT(1) was set to 2;
or $\theta(i)$, i=1 to N, if IOPT(1) was set to any other value.
N is the number of nodes.

After these data have been read from the disk file, and after the input data have been echoed to the default device, the prompt:

input: rain(L/T);end of rain(T);cond:

will be displayed on the default device. Respond by entering on the keyboard a value for:

RAIN = the precipitation rate for the period;
RAINTO= the end of the period:
COND = a): if COND = 9999, then the bottom node is considered a constant pressure node;
b):if COND has any value but 9999, then that is the constant flow into or out of the bottom of the column. Note that a positive value means upward flow into the bottom node; most of the time this will be a negative number, meaning water is draining out of the bottom node.

When the program has completed the period specified, the prompt will be repeated and the values of RAIN, RAINTO and COND can be repeated; note that RAINTO is the cumulative time, not the time interval of the next period. When you want to end the program, simply respond to the prompt with:

0 0 0

OUTPUT

On output, all of the input data is echoed back: the two lines of TITLE, followed by the parameters of input record #4, in the same order as they were input, but displayed on 2 lines. This is followed by a line with the IOPT parameters, a line with the IND parameters, a line with the WETP parameters and, if IOPT(3) is not 0, the tables of θ, K and ϕ for the different soil types.

Only if IOPT(1)=1, e.g. if the initial state is calculated from the given value of QS, will a table be printed of the initial values of θ, K and ϕ in the soil column. This table will contain two other items, labelled "It" and "UTES"; "It" is the number of iterations needed to achieve a ratio of < UTEST in the iterative routine for calculation of initial state; and "UTES" is the ratio actually achieved. It should be noted here that the initialization routine is not foolproof, and may not always converge: If

It > NMAX and UTES > UTEST

then conversion to the specified limit has not occurred. Nonconvergence however does not mean that the remainder of the program is invalid; it often will recuperate and converge to a stable state even from an unrealistic initial condition. In that case the user should perhaps consider running the program for a period in which a steady state or semi-steady state is achieved and take that steady state as the input to the simulation which is really wanted. This means that the time-variant part of the program is used to calculate the initial condition, rather than the steady state part of the program. Some experimentation will soon get you on the right track and a wizard at this procedure.

As an example, output from a short run is shown here. The text in { } are comments added for the purpose of this report, and are not part of the output.

EXAMPLE OUTPUT FILE

```
LAURA'S TANK;trial with inclusion of saturated flow:MOD BEUVING SND
20 ELEMENTS OF 20 CM;2M SAND,1M WASTE ,1M SAND ON TOP.
 20   20.00 10  .100E-03  .500E-02  .800E-02  .000E+00  3 18
  .100E+03  .000E+00 10     .05      .10
  1  2  1        {the IOPT-codes}
[the next line are IND showing the soil type for each element]
  3  3  3  3  3  3  3  3  3  3  2  2  2  2  2  3  3  3  3  3
nodes to be displayed:  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
  16 17 18 19 20  [display results for all the nodes]


soil type #  1          θ            K(cm/day)            ϕ(cm)
{                                                              }
                .85000E-01          .10000E-05         -.15849E+05
{this           .94000E-01          .30000E-05         -.10000E+05
 is the         .99000E-01          .50000E-05         -.79430E+04
 alternate      .10600E+00          .10000E-04         -.56230E+04
 soil}          .12000E+00          .33000E-04         -.31620E+04
                .13500E+00          .10600E-03         -.17780E+04
                .15100E+00          .34600E-03         -.10000E+04
                .16000E+00          .55500E-03         -.79400E+03
                .17100E+00          .11270E-02         -.56200E+03
                .19000E+00          .36310E-02         -.31600E+03
                .21500E+00          .24660E-01         -.17700E+03
                .25700E+00          .27040E+00         -.10000E+03
                .35200E+00          .91620E+00         -.56200E+02
                .40600E+00          .12650E+01         -.31600E+02
                .42500E+00          .21280E+01         -.10000E+02
                .42700E+00          .33270E+01         -.31600E+01
                .42900E+00          .51050E+01          .00000E+00
                .42920E+00          .51050E+01          .20000E+02
```

soil type # 2	θ	K(cm/day)	ϕ(cm)
{this is the waste}	.50000E-01	.10000E-05	-.15849E+05
	.90000E-01	.10000E-04	-.10000E+05
	.95000E-01	.40000E-04	-.79430E+04
	.10800E+00	.70000E-04	-.56230E+04
	.13900E+00	.10000E-03	-.31620E+04
	.17000E+00	.20000E-03	-.17780E+04
	.20700E+00	.50000E-03	-.10000E+04
	.22300E+00	.80000E-03	-.79400E+03
	.24500E+00	.10500E-02	-.56200E+03
	.27400E+00	.14400E+00	-.31600E+03
	.30000E+00	.23210E+01	-.17700E+03
	.32500E+00	.10830E+02	-.10000E+03
	.34000E+00	.26000E+02	-.56200E+02
	.35500E+00	.42520E+02	-.31600E+02
	.37600E+00	.65500E+02	-.10000E+02
	.37800E+00	.75100E+02	-.31600E+01
	.37900E+00	.80000E+02	.00000E+00
	.37910E+00	.80000E+02	.20000E+02

soil type # 3	θ	K(cm/day)	ϕ(cm)
{this is the soil used in the example run}	.50000E-02	.10000E-06	-.15849E+05
	.52000E-02	.10000E-05	-.10000E+05
	.54000E-02	.12000E-05	-.79430E+04
	.57000E-02	.15000E-05	-.56230E+04
	.60000E-02	.40000E-05	-.31620E+04
	.65000E-02	.11000E-04	-.17780E+04
	.70000E-02	.31000E-04	-.10000E+04
	.75000E-02	.46000E-04	-.79400E+03
	.90000E-02	.85000E-04	-.56200E+03
	.17000E-01	.23400E-03	-.31600E+03
	.37000E-01	.10470E-02	-.17700E+03
	.92000E-01	.19050E+00	-.10000E+03
	.22600E+00	.14490E+01	-.56200E+02
	.24200E+00	.23660E+01	-.31600E+02
	.24800E+00	.79980E+01	-.10000E+02
	.25000E+00	.27540E+02	-.31600E+01
	.25100E+00	.39880E+02	.00000E+00
	.25110E+00	.39880E+02	.20000E+02

downward flow= .2510 {the outflow from the bottom of the tank}
{the steady initial state which has been calculated by the model}
{note the non-convergence at nodes 8, 9, 10 and 17}

#	θ	K(CM/D)	ϕ(CM)	It	UTES
2	.24528E+00	.54483E+01	-.19779E+02	2	.13565E-04
3	.23754E+00	.21106E+01	-.38450E+02	3	-.10166E-05
4	.22636E+00	.14697E+01	-.55646E+02	3	-.52078E-04
5	.17891E+00	.10068E+01	-.71592E+02	4	.13143E-04
6	.13672E+00	.61048E+00	-.85383E+02	5	-.53215E-04
7	.10786E+00	.33950E+00	-.94814E+02	9	-.78748E-04
8	.99824E-01	.26398E+00	-.97443E+02	11	-.10709E+00
9	.10637E+00	.32548E+00	-.95302E+02	11	-.71764E+00
10	.10044E+00	.26976E+00	-.97242E+02	11	-.17068E+00
11	.31975E+00	.90438E+01	-.11616E+03	3	-.23199E-05
12	.31346E+00	.69033E+01	-.13553E+03	2	.44206E-04
13	.30725E+00	.47880E+01	-.15468E+03	3	-.13943E-05
14	.30119E+00	.27256E+01	-.17334E+03	3	-.21607E-04
15	.29733E+00	.20977E+01	-.19126E+03	3	-.68194E-06
16	.32759E-01	.87461E-03	-.20647E+03	2	.39931E-06
17	.25110E+00	.39880E+02	.20000E+02	11	-.52534E+05
18	.25100E+00	.39880E+02	.12588E+00	1	.00000E+00
19	.24532E+00	.54811E+01	-.19653E+02	2	.13397E-04
20	.23762E+00	.21151E+01	-.38331E+02	3	-.10145E-05

{This is the prompt for input of rainfall data. Since output was redirected to a disk file for use in this report, the prompt was similarly redirected.}

input:rain(L/T);end of rain(T);cond:
for period ending at time= .01 precip is .00

```
node #1 always at initial pressure: .0 {indicating the boundary cond.}

TIME:     .008 CHECKSUM: -.616256E-05 dT: .493753E-05
{a large checksum would indicate unacceptable loss of accuracy}

{The following lines are the output of results for one printstep}
{TOT.PRECIP= Total accumulated precipitation}
{POOL      = Height of water pooled on the surface}
{DITCH     = Accumulated outflow from the bottom}
{Σ θ*DZ    = Total amount of moisture in the column}
{INFILT    = Infiltration rate at end of step}
{GW. INFLOW= outflow (if negative) from the bottom of the tank}

TOT. PRECIP.        POOL      DITCH     Σ θ*DZ      INFILT.  GW. INFLOW
.000000E+00 .997284E+02 .200800E-02 .852116E+02 .339459E+02-.251000E+00

{and these are the moisture contents( volume fractions) for each of}
{the nodes                                                          }

.25100E+00  .24528E+00  .23754E+00  .22636E+00  .17891E+00  .13672E+00
.10787E+00  .99850E-01  .10635E+00  .10034E+00  .31984E+00  .31346E+00
.30725E+00  .30119E+00  .29723E+00  .13105E+00  .15290E+00  .25100E+00
.24532E+00  .25110E+00

 input:rain(L/T);end of rain(T);cond:
Stop - Program terminated.
```

Note that in the table showing the initial state no convergence to within the set limit was achieved for nodes #8, #9, #10 and #17, since for those nodes "It" > 10, the value specified for NMAX. The initialization routine has calculated very low values of θ for the first three nodes, whereas node #17 is set to saturation.

However, none of these unsatisfactory values is unacceptable to the time-variant portion of the program, which merrily goes on to calculate the state at the next timestep.

Not shown here is the case where the time-variant portion of the program does not converge in the iterative routine for the calculation of FB, the pressure at the boundary between two dissimilar nodes. The maximum number of iterations for this routine is set by the variable IT (record #4 of the input file). If no convergence is achieved in IT iterations, the last iteration is accepted as valid, but an error message is printed:

no convergence in fb-loop,node #, <NODE>, time=,<TIME>, Fbtst=,<FBT>

where <NODE> and <TIME> are the node and the time at which the non-convergence happened. <FBT> is the value of the achieved difference between the new and the previous value of FB, the pressure at the interface between the two nodes. The value aimed for is a difference of < 1 length unit.

The last few lines of the sample output show the data printed at intervals of DPRINT time units:

TIME = the total accumulated time;

CHECKSUM= is a test-value showing numerical dispersion up to that point in time. It is the total amount of water in the system minus the total amount of water at the beginning of the run. This number should ideally be zero, but values of up to .1 should not be considered bad for long time intervals where the total amounts of water passed through the system may well run into the 100's of length-units.

DT = the calculated timestep.

Then follows a line displaying:

The total precipitation to date;
The water accumulated in the pool;
The water accumulated in the ditch;
The total moisture in the soil column;
The infiltration rate at the end of the period;
The flow rate into(+) or out of(-) the bottom element.

And finally the moisture contents, in volume fractions, for each of the elements specified.

APPENDIX B

FORTRAN SOURCE CODE

```
$nofloatcalls
      program tankflo
c This version of tankflo incorporates the calculation of a
c maximum timestep each iteration; after "MAX" is calculated
c the going timestep is tested against "MAX" and if not in
c the range of [downrate*MAX - uprate*MAX] dta is reset to
c [uprate*MAX]. However, when either [downrate < 0 ] or
c [ uprate > 1.0 ] the routine for calculating  the maximum t-step
c will be bypassed entirely.
C CORRECTION 2-2-1987 ALSO INCLUDES TAKING OUT THE ROUTINE FOR
C FINDING INITIAL STATE FROM QS BY WAY OF FIRST CALCULATING THE
C PRESSURE AT THE BOUNDARY, AND REPLACING IT WITH FLOW BETWEEN
C SUCCESSIVE NODES, A BIT LESS ACCURATE, BUT LESS ERROR PRONE.
      COMMON TABLE(50,3,5),TARG(3),ISTART(50)
      DIMENSION WET(50), PER(50), FI(50)
      DIMENSION IOPT(3), IND(50), Q(50),PWET(30)
       CHARACTER*79 TXT
      INTEGER*2 WETP(30)
      READ (1,46) TXT
      WRITE(*,47) TXT
       READ(1,46) TXT
      WRITE(*,47) TXT
      READ (1,*) NUMSIM
      DO 44 KK=1,NUMSIM
      READ(1,*)N,DZ,nmax,utest,dprint,dt,time,mt,mte,pool,ditch,it
     1,downrate,uprate
      WRITE(*,101)N,DZ,nmax,utest,DPRINT,DT,TIME,MT,MTE,POOL,DITCH,it
     1,downrate,uprate
      READ (1,*) (IOPT(I),I=1,3)
      READ (1,*) (IND(I),I=1,N)
      READ(1,*)IDISP,(WETP(I),I=1,IDISP)
      read(1,*)(((table(i,j,k),i=1,mte),j=1,3),k=1,mt)
      WRITE(*,102 )(IOPT(I),I=1,3)
```

```
      WRITE(*,102)(IND(I),I=1,N)
      write(*,111)(wetp(i),i=1,idisp)
      if(iopt(3).eq.0)goto 66
      DO 1 K=1,MT
      WRITE(*,103) K
1     WRITE(*,58) ((TABLE(I,J,K),J=1,3),I=1,MTE)
C
C*********************************************************************
C
66    dprint=dprint*.9999
      DO 62 I=1,N
62    ISTART(I)=MTE/2
      dzs=dz*dz
C  SETTING UP OF INITIAL MOISTURE,PRESSURE AND CONDUCTIVITY PROFILES
C******
      IF (IOPT(1).EQ.1) GO TO 5
      IF (IOPT(1).EQ.2) GO TO 3
      READ (1,*) (WET(I),I=1,N)
      DO 2 I=1,N
      TARG(1)=WET(I)
      CALL LOOKUP (1,IND(I),MTE,I)
      PER(I)=TARG(2)
2     FI(I)=TARG(3)
      GO TO 61
3     READ (1,*) (FI(I),I=1,N)
      DO 4 I=1,N
      TARG(3)=FI(I)
      CALL LOOKUP(3,IND(I),MTE,I)
      PER(I)=TARG(2)
4     WET(I)=TARG(1)
      GO TO 61
5     READ (1,*) QS
      write(*,104)QS
```

```
      WRITE(*,115)
      FI(1)=0.
      targ(3)=0.
      ISTART(1)=MTE-2
      call lookup(3,ind(1),mte,1)
      wet(1)=targ(1)
      per(1)=targ(2)
      DO 14 I=2 ,N
      NTEST=0
      TARG(3)=FI(I-1)-DZ
      CALL LOOKUP(3,IND(I),MTE,I)
      AKNEW=TARG(2)
16    TARG(3)=FI(I-1)-DZ+2.*QS*DZ/(PER(I-1)+AKNEW)
      CALL LOOKUP(3,IND(I),MTE,I)
      UTES=(AKNEW-TARG(2))/AKNEW
      NTEST=NTEST+1
      IF(NTEST.GT.NMAX)GOTO 18
      IF(ABS(UTES).LT.UTEST)GOTO 18
      AKNEW=TARG(2)
      GOTO16
18    WET(I)=TARG(1)
      PER(I)=TARG(2)
      FI(I)=TARG(3)
14    WRITE(*,100)I,WET(I),PER(I),FI(I),NTEST,UTES
61    prnt=0.
      sum=0.
      do 59 i=1,n
59    sum=sum + wet(i)
      sum=sum*dz
      precip=0.
      TOTAL= DITCH + POOL + SUM
```

```
c=========
c==== start stepping through time  =======
c=========
17    write(*,*)' input:rain(L/T);end of rain(T);cond:'
      read(*,*)rain,tmax,cond
      tmax=tmax+.003
      if(tmax.le.time) goto 44
      write(*,105)tmax,rain
      if(cond.eq.9999.)write(0,112)fi(1)
      if(cond.ne.9999.)write(0,113)cond
      iptest=0
22    time=time+dt
      dta=dt
      if(time.lt.tmax)goto 23
      dta=tmax+dta-time
      time=tmax
      iptest=1
23    prnt=prnt+dta
      pool=pool+rain*dta
      precip=precip-rain*dta
      do 28 i=2,n
      if(ind(i-1).eq.ind(i))goto 67
      targ(3)=per(i)*(fi(i)+dz/2.)+per(i-1)*(fi(i-1)-dz/2.)
      targ(3)=targ(3)/(per(i-1)+per(i))
      itcnt=0
24    call lookup(3,ind(i),mte,I)
      itcnt=itcnt+1
      akup=targ(2)
      call lookup(3,ind(i-1),MTE,I-1)
      akd=targ(2)
      fb=(akup+per(i))*(fi(i)+dz/2.)
      fb=fb+(akd+per(i-1))*(fi(i-1)-dz/2.)
      DENOM= PER(I) + PER(I-1) + AKUP + AKD
      IF(ABS(DENOM). LT. 1.E-10) DENOM= 1.E-10
```

```
      fb=fb/DENOM
      if(fb.gt.table(mte,3,ind(i-1)))fb=table(mte,3,ind(i-1))
      fbtst=fb-targ(3)
      if(itcnt.lt.it)goto 25
      write(*,114) i,time,fbtst
      goto 26
25    if(abs(fbtst).le.1.)goto 26
      targ(3)=fb
      goto 24
26    ff=dta
      q(i)=-(per(i)+akup)*((fi(i)-fb)/dz+.5)
      goto 28
67    q(i)=dta*(per(i)+per(i-1))/2.
      ff=(fi(i-1)-fi(i))/dz-1.
28    q(i)=q(i)*ff
      q(1)=q(2)
      if(abs(cond-9999.).gt.1.e-6)q(1)=cond*dta
      q(n+1)=0.
      do 29 i=1,n
      add=(q(i)-q(i+1))/dz
      full=table(mte,1,ind(i))-.00001
      if(add+wet(i).lt.full)goto 29
      add=full-wet(I)
      q(i+1)=q(i)-add*dz
29    wet(i)=wet(i)+add
      full=table(mte,1,ind(n))
      dmax=(full-wet(n))*dz
      topk=table(mte,2,ind(n))
      qinf=(per(n)+topk)*(.5-fi(n)/dz)*dta
      if(qinf.gt.dmax)qinf=dmax
      IF(QINF.GT.POOL)QINF=POOL
      wet(n)=wet(n)+qinf/dz
      pool=pool-qinf
      ditch=ditch-q(1)
```

```
      DO 37 I=1,N
      targ(1)=wet(i)
      CALL LOOKUP(1,IND(i),MTE,I)
      FI(I)=TARG(3)
37    PER(I)=TARG(2)
      if(downrate.lt. 0.0 .or. uprate.gt.1.0) goto 72
      dtm=1.e12
      ILOOP=1
      IF(ABS(COND-9999.).LT.1.E-6)ILOOP=2
      IF(FI(1).GE.0.)ILOOP=2
      do 71 i=ILOOP,N
      targ(1)=wet(i)-.0001
      call lookup(1,ind(i),mte,I)
      hlo=targ(3)
      targ(1)=wet(i)+.0001
      call lookup(1,ind(i),mte,I)
      DENOM=HLO-TARG(3)
      IF(ABS(DENOM).LT.1.E-10)DENOM=1.E-10
      dtl=abs(dzs*.0001/(per(i)*DENOM))
      if(dtl.lt.dtm)dtm=dtl
71    continue
      up=uprate*dtm
      down=downrate*dtm
      if(dt.gt.up.or.dt.lt.down)dt=up
      if(dt.lt.1.e-6)STOP 'DT TOO SMALL'
72    IF(PRNT.LT.DPRINT)GOTO 42
      sum=0.
      do 63 i=1,n
63    sum=sum+wet(i)
      sum=sum*dz
      PRNT=0.
      qinf=qinf/dta
      q(1)=q(1)/dta
      sumsum=sum +pool+precip+ditch - TOTAL
```

```
      WRITE(*,106)TIME,SUMSUM,DT,PRECIP,POOL,DITCH,SUM,QINF,Q(1)
      DO 220 I=1,IDISP
220   PWET(I)=WET(WETP(I))
      WRITE(*,107)(PWET(I),I=1,IDISP)
42    IF(IPTEST.EQ.1)GOTO 17
      GOTO 22
44    CONTINUE
      stop
46    format(a79)
47    format(1x,a79)
58    format(7x,3e20.5)
100   FORMAT(1x,i4,3e12.5,i4,e12.5)
101   format(1x,i3,f8.2,i3,4e10.3,2i3/1x,2e10.3,i3,2f8.2)
102   format(1x,25i3)
103   format(1x,'soil                                                    type
#',i3,11X,'θ',12X,'K(cm/day',15x,'φ(cm)')
104   format(1x,'downward flow=',f10.4)
105   format('0for period ending at time=',f6.2,' precip is ',f6.2//)
106   FORMAT('0TIME:',F8.3,' CHECKSUM:',E13.6,' DT:',E12.6/
     13X,'TOT. PRECIP.',8X,'POOL',7X,'DITCH',6X,'Σ θ*DZ',5X,
     2'INFILT.  GW. INFLOW'/3X,6E12.6)
107   FORMAT(7X,6E12.5)
111   format(' nodes to be displayed:',15i3/(3x,20i3))
112   format(' node #1 always at initial pressure:',f10.1)
113   format(' constant outflow(-) or inflow(+) into node #1:',g12.4)
114   format(' no converge in fb-loop,node #',i3,
     1', time=',g11.4,'. Fbtst=',g12.4)
115   FORMAT(4X,'#',11X,'θ
K(CM/D)',7X,'φ(CM)',2X,'It',8X,'UTES')
      end

      SUBROUTINE LOOKUP (II,JJ,MTE,JS)
C*********************************************************************
C
```

```
C  LOOKUP INTERPOLATES BETWEEN ENTRIES OF TABLE TO FIND
C  CORRESPONDING VALUES OF TWO OF THE TABLE ARGUMENTS,
C  TARG,GIVEN THE THIRD ONE!THESE ARGUMENTS,AS WELL AS
C  THE TABULATED VALUES,ARE PASSED THROUGH COMMON.THE
C  SUBROUTINE ARGUMENTS ARE:
C     II=1,2,OR 3!THE INDEX OF TARG(II)!TARG(II) IS THE
C        TABLE ARGUMENT WITH THE GIVEN VALUE.
C     JJ=THE INDEX IND(SEE MAIN PROGRAM) INDICATING THE
C        SOIL TYPE FOR THE INTERPOLATION.
C     MTE=THE NUMBER OF TABLE ENTRIES.
C
C*********************************************************************
      COMMON TABLE(50,3,5),TARG(3),ISTART(50)
      P=TARG(II)
      TEST=P-TABLE(ISTART(JS),II,JJ)
      IF(TEST)1,2,3
2     IK=ISTART(JS)
7     DO 4 I=1,3
4     TARG(I)=TABLE(IK,I,JJ)
      RETURN
1     DO 5 I=2,ISTART(JS)
      JQ=ISTART(JS)+1-I
      IF(P.GT.TABLE(JQ,II,JJ))GOTO 6
5     CONTINUE
      IK=1
      ISTART(JS)=JQ
      GOTO 7
6     FAC=(P-TABLE(JQ,II,JJ))/(TABLE(JQ+1,II,JJ)-TABLE(JQ,II,JJ))
      DO 8 J=1,3
8     TARG(J)=TABLE(JQ,J,JJ)+FAC*(TABLE(JQ+1,J,JJ)-TABLE(JQ,J,JJ))
      ISTART(JS)=JQ
      RETURN
3     DO 9 I=ISTART(JS)+1,MTE
      IF(P.LT.TABLE(I,II,JJ))GOTO 10
```

```
9     CONTINUE
      IK=MTE
      ISTART(JS)=MTE
      GOTO 7
10    FAC=(P-TABLE(I-1,II,JJ))/(TABLE(I,II,JJ)-TABLE(I-1,II,JJ))
      DO 11 J=1,3
11    TARG(J)=TABLE(I-1,J,JJ)+FAC*(TABLE(I,J,JJ)-TABLE(I-1,J,JJ))
      ISTART(JS)=I
      RETURN
      END
```

APPENDIX C.

THE EXAMPLE DATA SET
FILE TANKTEST.DAT

This appendix is a copy of file TANKTEST.DAT, which contains data for a demonstration run of program TANKFLO.

```
Text in {  } has been inserted for this report only, and are not part
of the data on file TANKTEST.DAT


{ 2 lines of characters, the title}

LAURA'S TANK;trial with inclusion of saturated flow:MOD BEUVING SND
20 ELEMENTS OF 20 CM;2M SAND,1M WASTE ,1M SAND ON TOP.
{                                                                  }
{record #3: note that comments can be added on the same line,      }
{they will not be read by the program                              }
1    = numsim.NOTE THAT TABLE HAS BEEN REVISED AT THE WET END
{                                                                  }
{record #4: the 14 parameters                                      }
20 20 10 .0001 .0100 .0002  0 3 18 100 0 10  .95 .99
{                                                                  }
{record #5: the IOPT parameters                                    }
1 2 0
{                                                                  }
{record #6: the 20 indices, one for each element                   }
10*3 5*2 5*3
{                                                                  }
{record #7: the 20 indices of nodes for which moisture is printed  }
20 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
{                                                                  }
{record #8: the 18 table values, type 1, moisture content          }
.085 .094 .099 .106 .120 .135 .151 .160 .171 .190 .215 .257 .352
.406 .425 .427 .429 .4292
{record #8: 18 values of K }
.000001 .000003 .000005 .00001 .000033 .000106 .000346 .000555
.001127 .003631 .02466  .2704 .9162 1.265 2.128 3.327 5.105 5.105
{record #8: 18 values of pressure}
-15849 -10000 -7943 -5623 -3162 -1778 -1000 -794 -562 -316 -177 -100
-56.2 -31.6 -10 -3.16 0 20
```

```
{record #8: the same for soil types #2 and #3}
.050 .090 .095 .108 .139 .170 .207 .223 .245 .274 .300 .325 .340
.355 .376 .378 .379 .3791
.000001 .00001 .00004 .00007 .0001 .0002 .0005 .0008 .00105 .144
2.321 10.83 26.0 42.52 65.5 75.1 80 80
-15849 -10000 -7943 -5623 -3162 -1778 -1000 -794 -562 -316 -177 -100
-56.2 -31.6 -10 -3.16 0 20
.0050 .0052 .0054 .0057 .006 .0065 .007 .0075 .009 .017 .037 .092
.226 .242 .248 .250 .251 .2511
1.e-7 1.e-6 1.2e-6 1.5e-6 4e-6 11e-6 31e-6 46e-6 85e-6
.000234 .001047 .1905 1.449 2.366 7.998 27.54 39.88 39.88
-15849 -10000 -7943 -5623 -3162 -1778 -1000 -794 -562 -316 -177 -100
-56.2 -31.6 -10 -3.16 0 20
{                                                                    }
{and finally record #9: QS, the initial steady state flow rate       }
.251
```